

1 WHAT IS CLAIMED IS:

1 1. A method of assigning logical ports to respective peripheral devices
2 physically connected to a server in a network having client computers with client
3 software, wherein the respective peripheral devices are connected to one or more physical
4 port interfaces, the assigned logical ports enabling client software to communicate with
5 the peripheral devices regardless of the particular physical port interface the peripheral
6 device is connected to, the method comprising the steps of:

7 the server querying, responsive to a predetermined event, a peripheral device that
8 is connected to the server by a physical port interface for information which identifies the
9 peripheral device;

10 the peripheral device communicating said identification information to the server;

11 the server initially assigning a logical port identification for a peripheral device
12 that is connected to a physical port interface of the server and storing said logical port
13 identification and said identification information in memory;

14 the server thereafter directing communications for respective peripheral devices
15 from client computers to said assigned logical port for such peripheral devices, wherein
16 the communications are executed utilizing the client software.

1 2. A method as defined in claim 1 wherein the peripheral devices are
2 connected to the server via a physical parallel port interface.

1 3. A method as defined in claim 1 wherein the peripheral devices are
2 connected to the server via a physical Universal Serial Bus interface.

1 4. A method as defined in claim 1 wherein said predetermined event is the
2 initialization of the server.

1 5. A method as defined in claim 1 wherein said predetermined event is the
2 connection of a previously unconnected peripheral device to the server.

1 6. A method as defined in claim 1 wherein said peripheral device
2 identification information comprises one or more of the manufacturer, model name,
3 model number and serial number of the peripheral device.

1 7. A method as defined in claim 1 wherein said logical port identification and
2 said identification information are stored in a table having a predetermined maximum
3 number of logical port entries.

1 8. A method as defined in claim 7 wherein the status of each of said maximum
2 logical port entries is maintained in said table, with logical ports that are assigned having
3 an assigned status and logical ports that are unassigned having a free status.

1 9. A method as defined in claim 8 wherein when the server, responsive to said
2 predetermined event, queries peripheral devices that are then connected to the server,
3 compares said identification information received therefrom with said information
4 maintained in said table and utilizes said assigned logical ports for directing
5 communications to the connected peripheral devices in response to a positive
6 comparison.

1 10. A method as defined in claim 8 wherein when the server, responsive to said
2 predetermined event, queries peripheral devices that are then connected to the server,
3 compares said identification information received therefrom with said information
4 maintained in said table and fails to detect a positive comparison, performs the following
5 steps with respect to each peripheral device for which a positive comparison is not found:

6 the server searches the table to determine if the number of logical port entries is
7 less than the maximum number of entries, indicating that one or more logical port entries
8 are available;

9 the server assigns an available logical port identification to an unassigned
10 peripheral device and stores said logical port identification, said identification
11 information and an assigned status in an entry in the table in response to a logical port
12 being available;

13 the server suspends processing of the unassigned peripheral device for a
14 predetermined time period if there is no logical port available;

15 the server resumes processing after said predetermined time period and searches
16 said table for reserved status entries and selects a reserved entry that closely matches,

17 according to predetermined criteria, the identification information of the peripheral
 18 device that is unassigned and assigns the unassigned peripheral device to the logical port.

1 11. A method as defined in claim 10 wherein unassigned peripheral devices are
 2 placed in a queue for subsequent processing.

1 12. A method as defined in claim 1 wherein said memory is a non-volatile
 2 memory.

1 13. A method of assigning logical ports to respective peripheral devices
 2 physically connected to a server in a network having client computers with client
 3 software, wherein the respective peripheral devices are adapted to be connected to one or
 4 more physical port interfaces, the assigned logical ports enabling client software to
 5 communicate with the peripheral devices regardless of the particular physical port
 6 interface the peripheral device is connected to, the method comprising the steps of:

7 the server querying, responsive to a predetermined event, a peripheral device that
 8 is connected to the server by a physical port interface for information which identifies the
 9 peripheral device;

10 the peripheral device communicating said identification information to the server;

11 the server placing the unassigned peripheral devices in a queue for subsequent
 12 assignment of logical ports by manual assignment;

13 the server storing assigned logical port identification and said identification
 14 information in memory;

15 the server thereafter directing communications for respective peripheral devices
 16 from client computers to said assigned logical port for such peripheral devices wherein
 17 said communications are executed utilizing the client software.

1 14. A method as defined in claim 13 wherein said logical port identification
 2 and said identification information are stored in a table having a predetermined maximum
 3 number of logical port entries, said logical ports that are assigned having an assigned
 4 status and logical ports that are unassigned having a free status.

1 15. A system of assigning logical ports to respective peripheral devices
 2 physically connected to a server connected in a network including client computers

3 having client software for controlling the operation of the peripheral devices, wherein the
 4 respective peripheral devices are connected to one or more physical port interfaces, the
 5 assigned logical ports enabling client software to communicate with the peripheral
 6 devices regardless of the particular physical port interface the peripheral device is
 7 connected to, the system comprising:

8 the server being adapted to query, responsive to a predetermined event, a
 9 peripheral device that is connected to the server by a physical port interface for
 10 information which identifies the peripheral device;

11 the peripheral device being adapted to communicate said identification
 12 information to the server;

13 the server being adapted to initially assign a logical port identification for a
 14 peripheral device that is connected to a physical port interface of the server and store said
 15 logical port identification and said identification information in memory;

16 the server thereafter directing communications for respective peripheral devices
 17 from client computers to said assigned logical port for such peripheral devices, wherein
 18 the client computers communicate with the peripheral device using the client software.

1 16. A system as defined in claim 15 wherein the server stores logical port
 2 identification and said identification information in a table having a predetermined
 3 maximum number of logical port entries.

1 17. A system as defined in claim 16 wherein the server stores the status of each
 2 of said maximum logical port entries in said table, with logical ports that are assigned
 3 having an assigned status and logical ports that are unassigned having a free status.

1 18. A system as defined in claim 17 wherein when the server, responsive to
 2 said predetermined event, queries peripheral devices that are then connected to the server,
 3 compares said identification information received therefrom with said information
 4 maintained in said table and directs communications to the connected peripheral devices
 5 using said assigned logical ports in response to a positive comparison.

1 19. A system as defined in claim 18 wherein when the server, responsive to
 2 said predetermined event, queries peripheral devices that are then connected to the server,

3 compares said identification information received therefrom with said information
4 maintained in said table and fails to detect a positive comparison, the server performs the
5 following steps with respect to each peripheral device for which a positive comparison is
6 not found:

7 the server searches the table to determine if the number of logical port entries is
8 less than the maximum number of entries, indicating that one or more logical port entries
9 are available;

10 the server assigns an available logical port identification to an unassigned
11 peripheral device and stores said logical port identification, said identification
12 information and an assigned status in an entry in the table in response to a logical port
13 being available;

14 the server suspends processing of the unassigned peripheral device for a
15 predetermined time period if there is no logical port available;

16 the server resumes processing after said predetermined time period and searches
17 said table for reserved status entries and selects a reserved entry that closely matches,
18 according to predetermined criteria, the identification information of the peripheral
19 device that is unassigned and assigns the unassigned peripheral device to the logical port.

1 20. A computer program product comprising a computer usable medium
2 having computer readable program codes embodied in the medium that when executed
3 causes the server to:

4 query a peripheral device that is connected to the server by a physical port
5 interface for information which identifies the peripheral device, responsive to a
6 predetermined event;

7 receive identification information that is transmitted by the peripheral device;

8 assign a logical port identification for the peripheral device that transmitted the
9 identification information and store said logical port identification and said identification
10 information in memory;

thereafter direct communications for the peripheral device from client computers to said assigned logical port for such peripheral devices, wherein the communications are executed utilizing the client software.

21. A product as defined in claim 20 which causes the server to store the status of each of said logical port entries in said table, with logical ports that are assigned having an assigned status and logical ports that are unassigned having a free status.

22. A product as defined in claim 21 which causes the server, when responsive to said predetermined event, to query peripheral devices that are then connected to the server, to compare said identification information received therefrom with said information maintained in said table and direct communications to the connected peripheral devices using said assigned logical ports in response to a positive comparison.

23. A product as defined in claim 22 which causes the server, when responsive to said predetermined event, to query peripheral devices that are then connected to the server, to compares said identification information received therefrom with said information maintained in said table and fails to detect a positive comparison, perform the following steps with respect to each peripheral device for which a positive comparison is not found:

search the table to determine if the number of logical port entries is less than a maximum number of entries, indicating that one or more logical port entries are available;

assign an available logical port identification to an unassigned peripheral device and store said logical port identification, said identification information and an assigned status in an entry in the table in response to a logical port being available;

suspend processing of the unassigned peripheral device for a predetermined time period if there is no logical port available;

resume processing after said predetermined time period and search said table for reserved status entries and select a reserved entry that closely matches, according to predetermined criteria, the identification information of the peripheral device that is unassigned and assign the unassigned peripheral device to the logical port.

24. A product as defined in claim 23 where said identification of information that approximates said identification information comprises at least identical manufacturer and model information.